

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A method for partitioning a large computer program at least part of which is to be executed by a reconfigurable array comprising an array of reconfigurable units including a plurality of Arithmetic Logic Units (ALUS) and a plurality of memory units, comprising the steps of:

defining a maximum allowable size to be mapped onto the reconfigurable array;  
and

partitioning the program such that:

its separate parts minimize the ~~overall~~ execution time; ~~and provide~~  
a mapping onto the reconfigurable array does not exceed ~~exceeding~~ the  
maximum allowable size; and  
an overhead for reconfiguring the reconfigurable array is minimized.

2. (Currently Amended) A device for partitioning a large computer program at least part of which is to be executed by a reconfigurable array comprising an array of reconfigurable units including a plurality of Arithmetic Logic Units (ALUS) and a plurality of memory units, comprising:

an arrangement for defining a maximum allowable size to be mapped onto the reconfigurable array; and

an arrangement for partitioning the program such that:

its separate parts minimize the ~~overall~~ execution time; ~~and provide~~  
a mapping onto the reconfigurable array does not exceed ~~exceeding~~ the  
maximum allowable size; and  
an overhead for reconfiguring the reconfigurable array is minimized.

3. (Currently Amended) A method for partitioning at least one of large computer programs and large algorithms at least part of which is to be executed by a reconfigurable array comprising an array of reconfigurable units, the array including a plurality of reconfigurable memory units, the method comprising:

defining a maximum allowable size to be mapped onto the array;

partitioning at least one of a program and an algorithm such that:

its separate parts (a) minimize an ~~overall~~ execution time of the at least one of the program and the algorithm and (b) each does not exceed the maximum allowable size; and

an overhead for reconfiguring the array is minimized; and  
mapping the separate parts onto the array.

4. (Previously Presented) The method of claim 3, wherein the array further includes a plurality of reconfigurable arithmetic logic units.

5. (New) A method for compiling a large computer program at least part of which is to be executed by a reconfigurable array comprising an array of reconfigurable units including a plurality of Arithmetic Logic Units (ALUS) and a plurality of memory units, the method comprising:

loading configurations to a cache according to at least one of the following strategies:

(1) a strategy according to which all configurations existent in an application are requested for pre-fetch at a start of the execution of the program; and

(2) a strategy according to which, in each configuration, a following configuration is requested for pre-fetch.

6. (New) The method of claim 5, further comprising:

defining a maximum allowable size to be mapped onto the reconfigurable array;

and

partitioning the program such that:

its separate parts minimize the execution time;

a mapping onto the reconfigurable array does not exceed the maximum allowable size; and

an overhead for reconfiguring the reconfigurable array is minimized.

7. (New) The method according to claim 6, wherein, for reducing the overhead, the partitions into which the program is partitioned are optimized by at least one of:

combining only most frequently taken conditional paths in a same partition; and  
implementing a loop entirely by a single configuration.

8. (New) The method according to claim 1, wherein, for reducing the overhead, the partitions into which the program is partitioned are optimized by at least one of:

combining only most frequently taken conditional paths in a same partition; and  
implementing a loop entirely by a single configuration.

9. (New) A method for partitioning a large computer program at least part of which is to be executed by an array of reconfigurable units including a plurality of Arithmetic Logic Units (ALUS) and a plurality of memory units, comprising the steps of:

defining a maximum allowable size to be mapped onto the array; and  
partitioning the program such that its separate partitioned parts (a) minimize an overall execution time of the program and (b) each does not exceed the maximum allowable size, the partitioned parts being mapped onto the plurality of ALUS and onto the plurality of memory units.